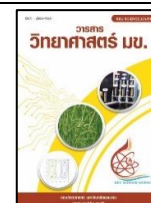




KKU SCIENCE JOURNAL

Journal Home Page : <https://ph01.tci-thaijo.org/index.php/KKUSciJ>

Published by the Faculty of Science, Khon Kaen University, Thailand



การเลือกคุณลักษณะและการจำแนกโรคความดันโลหิตสูงโดยใช้ขั้นตอนวิธีการหาค่าเหมาะที่สุดแบบความอลวนกับนกอินทรีทองคำ

Feature Selection and Classification of Hypertension Using the Chaotic Golden Eagle Optimization Algorithm

บุญญฤทธิ์ เพ็ญพูล¹ ธรรมณูญ ปัญญาทิพย์¹ และ ธนัชพงษ์ วงศ์คำหาญ^{1*}Punyarit Permpool¹, Tammanoon Panyatip¹ and Tanachapong Wangkhamhan^{1*}¹สาขาวิชาวิทยาการคอมพิวเตอร์และเทคโนโลยีสารสนเทศ คณะวิทยาศาสตร์และเทคโนโลยีสุขภาพ มหาวิทยาลัยกาฬสินธุ์ จังหวัดกาฬสินธุ์ 46000¹Computer Science and Information Technology, Faculty of Science and Health Technology, Kalasin University, Kalasin, 46000, Thailand.

บทคัดย่อ

โรคความดันโลหิตสูงเป็นปัญหาสาธารณสุขสำคัญระดับโลกและเป็นปัจจัยเสี่ยงหลักของโรคหัวใจและหลอดเลือด การจำแนกและตรวจคัดกรองผู้ป่วยได้อย่างถูกต้องตั้งแต่ระยะเริ่มต้นมีความสำคัญต่อการวางแผนรักษาและลดภาวะแทรกซ้อนในระยะยาว อย่างไรก็ตาม แบบจำลองการเรียนรู้ของเครื่องมักได้รับผลกระทบจากคุณลักษณะที่ซ้ำซ้อน ไม่เกี่ยวข้อง หรือมีความสัมพันธ์กันสูง ซึ่งอาจลดประสิทธิภาพการจำแนกและเพิ่มภาระเชิงคำนวณ งานวิจัยนี้นำเสนอขั้นตอนวิธีการหาค่าเหมาะที่สุดแบบความอลวนกับนกอินทรีทองคำ สำหรับการเลือกคุณลักษณะและการจำแนกโรคความดันโลหิตสูง โดยพัฒนาต่อยอดจากขั้นตอนวิธีการหาค่าเหมาะที่สุดนกอินทรีทองคำ ด้วยการผสมผสานความอลวนแบบ Chebyshev เพื่อเพิ่มความหลากหลายของประชากร ปรับสมดุลระหว่างการสำรวจกับการแสวงหาคำตอบ และลดปัญหาการลู่เข้าเร็วเกินไป วิธีที่เสนอถูกประเมินบนชุดข้อมูลความดันโลหิตสูงจริงซึ่งประกอบด้วยข้อมูลทางคลินิกและข้อมูลประชากร โดยผลการทำงานร่วมกับตัวจำแนก 4 แบบ ได้แก่ Support Vector Machine (SVM) k-Nearest Neighbors (KNN) Decision Tree (DT) และ Naïve Bayes (NB) ภายใต้การตั้งค่าการทดลองเดียวกัน ประสิทธิภาพของแบบจำลองประเมินด้วย Accuracy Precision Recall F1-score และ Specificity จากการทดลองอิสระ 30 ครั้ง ผลการทดลองพบว่า CGEO ให้ผลดีกว่า GEO อย่างสม่ำเสมอในทุกตัวจำแนก โดยแบบจำลอง CGEO-SVM ให้ผลดีที่สุด มีความแม่นยำร้อยละ 95.78 และค่าความจำเพาะร้อยละ 96.20 พร้อมทั้งสามารถลดจำนวนคุณลักษณะลงได้ประมาณร้อยละ 50 นอกจากนี้ การเปรียบเทียบเพิ่มเติมภายใต้ตัวจำแนก SVM กับขั้นตอนวิธี PSO GWO และ GA แสดงให้เห็นว่า CGEO ให้ความสมดุลโดยรวมที่ดีที่สุดในด้านความแม่นยำ ความจำเพาะ ค่า F1-score อัตราการลดคุณลักษณะ และเวลาในการประมวลผลภายใต้เงื่อนไขการทดลองปัจจุบัน การทดสอบทางสถิติแสดงให้เห็นว่าความแตกต่างระหว่าง GEO และ CGEO มีนัยสำคัญทางสถิติที่ระดับ 0.05 นอกจากนี้ แม้ CGEO จะมีเวลาในการประมวลผลเพิ่มขึ้นเล็กน้อย แต่ยังอยู่ในระดับที่ยอมรับได้เมื่อเทียบกับประโยชน์ที่ได้รับในด้านความแม่นยำ ความเสถียร และความกะทัดรัดของชุดคุณลักษณะ ผลการศึกษาชี้ให้เห็นว่าการผสม

*Corresponding Author, E-mail: tanachapong.w@hotmail.com

Received date: 7 November 2025 | Revised date: 29 April 2026 | Accepted date: 25 June 2026

doi: 10.14456/kkuscij.2026.36

กลไกเชิงโกลาหลเข้ากับการอบเมตาฮิวริสติกสามารถช่วยเพิ่มประสิทธิภาพของการเลือกคุณลักษณะและการจำแนกข้อมูลทางการแพทย์ได้อย่างมีนัยสำคัญ

ABSTRACT

Hypertension is a major global public health problem and one of the leading risk factors for cardiovascular disease. Accurate classification and early identification are essential for effective treatment planning and long-term risk reduction. However, machine-learning models are often affected by redundant, irrelevant, or highly correlated features, which may reduce classification performance and increase computational burden. This study proposes a Chaotic Golden Eagle Optimization (CGEO) algorithm for feature selection and hypertension classification. The proposed method extends the original Golden Eagle Optimizer (GEO) by integrating a Chebyshev chaotic map to improve population diversity, strengthen the exploration–exploitation balance, and reduce premature convergence. The method was evaluated on a real hypertension dataset containing clinical and demographic attributes and was integrated with four classifiers: Support Vector Machine (SVM), k-Nearest Neighbors (KNN), Decision Tree (DT), and Naïve Bayes (NB) under the same experimental settings. Model performance was assessed using accuracy, precision, recall, F1-score, and specificity over 30 independent runs. The experimental results showed that CGEO consistently outperformed GEO across all classifiers. Among the evaluated models, CGEO-SVM achieved the best overall performance, with an accuracy of 95.78 percent and a specificity of 96.20 percent, while reducing the number of selected features by approximately 50%. An additional controlled SVM-based benchmark against PSO, GWO, and GA further showed that CGEO achieved the most favorable overall trade-off in terms of accuracy, specificity, F1-score, feature-reduction rate, and runtime under the present experimental setting. Statistical testing further confirmed that the differences between GEO and CGEO were significant at the 0.05 level. In addition, although CGEO required slightly more computation time than GEO, the overhead remained acceptable relative to the gains in predictive performance, stability, and feature subset compactness. These findings indicate that incorporating chaotic dynamics into a metaheuristic framework can significantly improve feature selection and medical data classification performance.

คำสำคัญ: การจำแนกโรคความดันโลหิตสูง ขั้นตอนวิธีการหาค่าเหมาะที่สุดแบบความอลวนกับนกอินทรีทองคำ การเลือกคุณลักษณะ การเรียนรู้ของเครื่อง Chebyshev Map

Keywords: Hypertension Classification, Chaotic Golden Eagle Optimization, Feature Selection, Machine Learning, Chebyshev Map

INTRODUCTION

Hypertension, commonly referred to as high blood pressure, remains one of the most prevalent chronic diseases worldwide and is a major contributor to cardiovascular morbidity and mortality (World Health Organization, 2024). Because the disease often develops gradually and may remain asymptomatic for long periods, accurate classification and early identification are essential for timely intervention, effective treatment planning, and reduction of long-term complications. However, reliable hypertension classification

remains challenging because medical data are often characterized by nonlinear relationships, heterogeneous clinical patterns (Al-Gburi and Kurnaz, 2022), and complex interactions among physiological variables. In such contexts, redundant, weakly relevant, or highly correlated features may reduce model effectiveness, increase computational cost, and limit interpretability. In recent years, machine learning (ML) and artificial intelligence (AI) techniques have been increasingly adopted in medical diagnosis and prediction because they can learn complex relationships from multidimensional clinical data more effectively than many conventional statistical approaches. Nevertheless, the predictive performance of ML models depends strongly on the quality, relevance, and redundancy level of the input features. When irrelevant or highly correlated variables are included, classification models may become less accurate, less interpretable, and more computationally expensive. For this reason, feature selection (FS) has become an important component of medical data analysis (Liu *et al.*, 2019), particularly in diagnostic classification tasks where both predictive performance and practical interpretability are required. Among the available feature-selection approaches, metaheuristic algorithms have gained considerable attention because of their flexibility, adaptability, and global search ability in complex search spaces. Population-based and swarm-inspired methods such as Particle Swarm Optimization (PSO), Grey Wolf Optimizer (GWO), and the more recently proposed GEO have demonstrated competitive performance in solving nonlinear and high-dimensional optimization problems. GEO is inspired by the hunting behavior of golden eagles and models two complementary search phases (Mohammadi-Balani *et al.*, 2021), namely cruising and attacking, to balance exploration and exploitation within a unified optimization framework (Jayalekshmi and Velusamy, 2023). As discussed in The Original GEO Algorithm, this structure makes GEO attractive for wrapper-based feature selection because it offers a relatively simple search mechanism with limited control parameters and good optimization capability. Despite these strengths, the original GEO may still suffer from premature convergence and reduced population diversity when applied to highly correlated or multimodal search spaces (Eluri and Devarakonda, 2022). This limitation is especially relevant in wrapper-based feature selection, where many candidate feature subsets may exhibit similar local fitness values and the optimizer can become trapped around suboptimal regions. As explained in Limitations and Research Motivation, this issue reduces the ability of the algorithm to continue exploring alternative feature subsets and may negatively affect both predictive performance and subset compactness (Yang *et al.*, 2023). To address this limitation, chaotic theory has been incorporated into many metaheuristic algorithms as a mechanism for enhancing search dynamics. Chaotic sequences are deterministic, non-periodic, and highly sensitive to initial conditions, yet they exhibit pseudo-random and ergodic behavior over time. These properties make chaotic maps suitable for improving search-space coverage, maintaining diversity, and delaying premature convergence. Chaotic Maps further explains how chaos can act as a dynamic search-enhancement mechanism rather than as a separate optimization principle.

Although the use of chaotic maps in metaheuristic optimization is not entirely new, the integration of chaotic dynamics into the Golden Eagle Optimizer for wrapper-based feature selection in hypertension classification remains insufficiently explored. Previous studies have shown that chaotic enhancement can

be beneficial in other optimization frameworks, but limited attention has been given to how a specific chaotic map can be embedded into GEO to address diversity loss, improve search stability, and enhance the selection of classification-oriented feature subsets in medical datasets. Therefore, the novelty of this study does not lie merely in adding a chaotic component to an existing optimizer, but in specifically adapting the Chebyshev chaotic map within the GEO framework for a clinically relevant feature-selection problem.

The Chebyshev map is particularly suitable for this purpose because of its strong ergodicity, deterministic irregularity, and low computational overhead. Compared with conventional uniform random perturbation, Chebyshev-based chaotic sequences can generate more diversified and less repetitive search trajectories, especially during the early search stage, while still supporting effective refinement in later iterations. As detailed in Chebyshev Map, the map is normalized and integrated into the GEO movement mechanism. The resulting optimizer, called Chaotic Golden Eagle Optimization (CGEO), preserves the original hunting-inspired structure of GEO while improving its search dynamics through chaos-guided modulation. The actual implementation of this idea is presented mathematically and operationally in The Proposed CGEO Algorithm.

In this study, the proposed CGEO algorithm is applied to wrapper-based feature selection and hypertension classification by integrating it with four machine learning classifiers: SVM, KNN, DT, and NB. These classifiers were selected because they represent different learning paradigms and therefore provide a broader view of the usefulness of the selected feature subsets. The performance of CGEO is then compared with that of the standard GEO under identical experimental conditions, including the same dataset, preprocessing steps, classifier settings, validation protocol, and performance measures. The framework is evaluated using five widely used metrics, namely accuracy, precision, recall, F1-score, and specificity, over 30 independent runs to ensure statistical robustness and reproducibility, as described in Experimental Setup and Parameter Configuration. The major contributions of this paper can be summarized as follows:

1. A CGEO algorithm is developed to address the diversity loss and premature convergence limitations of the original GEO in wrapper-based feature selection.
2. The proposed CGEO embeds chaotic dynamics into the GEO search mechanism to improve search-space coverage and regulate the transition between exploration and exploitation.
3. A unified comparative framework is established by integrating CGEO with four different classifiers (SVM, KNN, DT, and NB) for hypertension classification under identical experimental conditions.
4. The study provides empirical evidence that CGEO yields more stable and accurate feature-selection outcomes than the standard GEO and further evaluates its competitiveness against PSO, GWO, and GA under a controlled SVM-based benchmark setting.

The remainder of this paper is organized as follows. The Original GEO Algorithm reviews the baseline optimization framework, including its conceptual model, mathematical formulation, parameter settings, applications, and limitations. Chaotic Maps introduces the role of chaotic maps in metaheuristic

optimization and explains why the Chebyshev map was selected for this study. The Proposed CGEO Algorithm presents the proposed approach, including its framework overview, mathematical formulation, fitness function design, and workflow. Experimental Setup and Parameter Configuration describes the dataset, preprocessing procedures, parameter settings, classifier configurations, evaluation metrics, and validation strategy. Results and Discussion reports and discusses the comparative experimental results in terms of overall performance, the impact of chaotic integration, statistical robustness, feature-selection efficiency, and clinical and computational significance. Finally, Conclusion and Future Work summarizes the study and outlines its theoretical implications, practical relevance, limitations, and directions for future research.

MATERIALS AND METHODS

1. The original GEO algorithm

The GEO is a nature-inspired metaheuristic algorithm proposed by (Mohammadi-Balani *et al.*, 2021) based on the hunting behavior of golden eagles. In nature, golden eagles exhibit two major behavioral modes during prey hunting: cruising, which corresponds to exploratory scanning over a broad region, and attacking, which corresponds to directed movement toward a promising target. These two complementary behaviors provide a natural analogy for the balance between exploration and exploitation, which are fundamental components of population-based optimization (Rakotonirainy and van Vuuren, 2020). In the present study, GEO serves as the baseline optimization framework upon which the proposed CGEO is later developed. Therefore, understanding the structure, movement mechanism, and limitations of the original GEO is important, as these elements directly motivate the discussions presented in Chaotic Maps, The Proposed CGEO Algorithm, and Experimental Setup and Parameter Configuration, while also providing a basis for interpreting the comparative findings reported in Results and Discussion and summarized in Conclusion and Future Work.

1.1 Conceptual framework

In GEO, each eagle represents a candidate solution in the search space, whereas the prey represents a promising or currently best-known solution. The search process is governed by two main movement tendencies (Mohammadi-Balani *et al.*, 2021):

- 1) Cruising (exploration): During cruising, the eagle searches around the prey region while maintaining enough distance to explore neighboring areas. This movement supports global search and helps the optimizer examine multiple candidate regions.

- 2) Attacking (exploitation): During attacking, the eagle moves more directly toward the prey. This behavior supports local refinement and helps the optimizer intensify the search around promising solutions.

The interaction between these two tendencies allows GEO to transition gradually from diversified exploration in the early search stage to intensified exploitation in later iterations. In optimization terms, this behavior enables the algorithm to search broadly at first and then refine candidate solutions as the

population converges (Heidari *et al.*, 2019). Conceptually, the GEO framework can be summarized as follows:

- each eagle = one candidate solution,
- the prey = the best solution found so far,
- cruise behavior = search-space exploration,
- attack behavior = solution refinement.

This conceptual structure is important because the proposed CGEO does not replace the GEO mechanism entirely. Instead, it preserves this hunting-inspired framework and later enhances it through chaotic modulation, as described in Chaotic Maps and The Proposed CGEO Algorithm.

1.2 Mathematical model

Let X_i^t denote the position of the i -th eagle at iteration t , and let X_p^t denote the position of the prey or current best solution. In GEO, the movement of each eagle is determined by two velocity components: the cruise velocity and the attack velocity. The cruise velocity can be expressed as Equation (1):

$$C_i^t = r_1 \cdot (X_p^t - X_i^t) \quad (1)$$

and the attack velocity can be expressed as Equation (2):

$$A_i^t = r_2 \cdot (X_p^t - X_i^t) \quad (2)$$

where r_1 and r_2 are random coefficients used to introduce stochasticity into the search process. The total movement of each eagle is then obtained by combining the two components, shown in Equation (3):

$$V_i^t = \alpha_t C_i^t + \beta_t A_i^t \quad (3)$$

where α_t and β_t are coefficients that regulate the relative influence of cruising and attacking behavior at iteration t . These coefficients are typically varied over time so that exploration dominates in the early phase and exploitation becomes stronger in the later phase. The position of each eagle is then updated as shown in Equation (4):

$$X_i^{t+1} = X_i^t + fl \cdot V_i^t \quad (4)$$

where fl is the flight-length coefficient controlling the movement step size. After each update, the fitness values of all eagles are evaluated, and the prey position is updated if a better solution is found. This iterative process continues until a stopping criterion is met, such as a maximum number of iterations or convergence of the best solution.

This mathematical structure forms the foundation of The Proposed CGEO Algorithm. In particular, the key modification introduced later is not a complete redesign of GEO, but rather the replacement or modulation of the stochastic coefficients by Chebyshev chaotic values, thereby preserving the same movement logic while enhancing the search dynamics.

1.3 Parameter settings and complexity

One of the practical strengths of GEO is that it uses a relatively small number of control parameters compared with many other population-based optimizers (Mohammadi-Balani *et al.*, 2021). In this study, the main parameters associated with GEO are:

- population size N ,
- maximum number of iterations T ,
- attack-related coefficient,
- cruise-related coefficient,
- and flight length fl .

The parameter configuration used in the experiments is reported in Table 2 under Parameter Settings for GEO and CGEO, where both GEO and CGEO are evaluated under the same basic settings to ensure a fair comparison. In the present work, the key values are:

- $N = 100$,
- $T = 100$,
- $fl = 0.7$,
- and dynamically adjusted cruise/attack propensities.

This consistency is important because the observed differences in performance, feature-selection efficiency, and runtime reported throughout Results and Discussion can be attributed to the chaotic enhancement introduced in CGEO rather than to unequal baseline settings. In terms of computational cost, the time complexity of GEO per iteration is typically proportional to $O(N \times d)$, where N is the population size and d is the dimensionality of the search space. In wrapper-based feature selection, the practical computational burden is also influenced by the cost of classifier training and validation, since each eagle must be evaluated through the classifier-dependent fitness function. For this reason, the runtime comparison presented in Clinical and Computational Significance provides an important complement to the classification results. Thus, GEO provides an attractive baseline because it combines (Yang *et al.*, 2023):

- conceptual simplicity,
- a limited number of tunable parameters,
- and computational efficiency that remains comparable to other population-based methods.

1.4 Applications for GEO

Since its introduction, GEO has been applied to a variety of optimization problems across different domains (Jayalekshmi and Velusamy, 2023). Previous studies have reported that GEO is effective in applications such as:

- clustering and network optimization,
- image segmentation (Al-Gburi and Kurnaz, 2022),
- engineering optimization,
- and feature selection for classification tasks (Eluri and Devarakonda, 2022).

These prior applications suggest that GEO is sufficiently flexible to operate in both continuous and discrete search spaces and can perform competitively in nonlinear optimization problems. Its balance between exploration and exploitation makes it suitable for feature-selection problems, where the optimizer must navigate a large combinatorial space of candidate subsets (Eluri and Devarakonda, 2022). In the context of the present study, GEO is especially relevant because the hypertension dataset contains:

- multiple correlated attributes,
- nonlinear interactions among features,
- and a search space in which many feature subsets may yield similar local fitness values.

This makes GEO an appropriate baseline algorithm for evaluating whether the proposed chaotic enhancement can improve search stability, subset compactness, and classifier performance. Moreover, the value of GEO in this article is not only methodological but also comparative. Because the experimental framework described in Experimental Setup and Parameter Configuration and the findings presented in Results and Discussion primarily focus on comparing GEO with CGEO, with an additional controlled SVM-based benchmark against PSO, GWO, and GA, the baseline behavior of GEO provides the primary reference point for interpreting the contribution of chaotic enhancement:

- the higher classification scores of CGEO,
- the lower variability of CGEO across 30 runs,
- the stronger feature-reduction efficiency of CGEO,
- and the moderate runtime overhead introduced by chaotic modulation.

1.5 Limitations and research motivation

Despite its effectiveness, the original GEO may still encounter difficulties in complex optimization tasks. The main limitations relevant to the present study are as follows (Yang *et al.*, 2023).

1) Premature convergence: When multiple candidate solutions cluster around a locally attractive region, the search may converge too early before sufficiently exploring alternative areas.

2) Loss of population diversity: In high-dimensional or highly correlated search spaces, the stochastic movement mechanism of GEO may not always generate enough variation to maintain broad search coverage.

3) Sensitivity in wrapper-based feature selection: In medical feature-selection problems, many feature subsets can have similar local fitness values. Under such conditions, a purely stochastic search process may become trapped near suboptimal subsets, reducing both classification quality and feature compactness.

These limitations are particularly relevant to the present hypertension classification problem. As demonstrated in Impact of Chaotic Chebyshev Map Integration, Statistical Robustness and Stability, and Feature-Selection Efficiency, the study seeks not only to improve classification accuracy but also to achieve:

- stronger repeated-run stability,
- better feature-selection consistency,
- and more compact feature subsets.

For this reason, the present study introduces chaotic dynamics into the GEO framework. The motivation is that a deterministic but irregular sequence, such as the Chebyshev chaotic map, may improve the diversification mechanism of GEO without changing its original search structure. This leads directly to the next section, where the role of chaotic maps in GEO is described in detail, shown in **Algorithm 1**:

Algorithm 1. The GEO algorithm (Boriratr et al., 2023).

- 1: Initialize the population size of eagles (N), maximum number of iterations (T), attack coefficient (α), cruise coefficient (β), and control parameter (λ)
- 2: Generate the initial population of eagles $\{X_i\}_{i=1}^N$ randomly within the search space
- 3: Evaluate the fitness value of all eagles and set the best one as the prey (elite solution)
- 4: **While** (termination criterion is not satisfied) **do**
- 5: **For** each eagle $i = 1 \rightarrow N$ **do**
- 6: Compute the cruise (exploration) velocity using

$$V_c = \beta \cdot r_1 \cdot (X_{\text{prey}} - X_i)$$
- 7: Compute the attack (exploitation) velocity using

$$V_a = \alpha \cdot r_2 \cdot (X_{\text{prey}} - X_i)$$
- 8: Combine both movements using the weighting parameter

$$V_i = \lambda \cdot V_c + (1 - \lambda) \cdot V_a$$
- 9: Update the position of eagle

$$X_i^{\text{new}} = X_i + V_i$$
- 10: Apply boundary control to keep X_i^{new} within the lower and upper limits
- 11: Evaluate the fitness value of the updated eagle
- 12: **End for**
- 13: Sort all eagles according to their fitness values
- 14: Update the prey (global best) if a better eagle is found
- 15: Update the control parameter λ (e.g. linearly decreasing) to shift from exploration to exploitation
- 16: **End while**
- 17: Output the best fitness value and the corresponding prey position

2. Chaotic maps for the GEO

Although the original GEO has demonstrated promising global search capability, its performance may still deteriorate in high-dimensional, highly correlated, or multimodal search spaces due to loss of population diversity and early convergence around suboptimal regions (Eluri and Devarakonda, 2022; Yang et al., 2023). This limitation becomes particularly critical in wrapper-based feature selection, where many candidate feature subsets may have similar local fitness values and the search process can easily become trapped around locally attractive solutions. In medical datasets, this difficulty is often amplified by feature redundancy, nonlinear dependency among variables, and noise in clinical measurements. For these reasons, enhancing the diversification mechanism of GEO is important for improving its search stability and optimization effectiveness. A widely adopted strategy in population-based optimization is to incorporate chaotic dynamics into the algorithmic search process. Chaotic sequences are deterministic, non-periodic,

and highly sensitive to initial conditions, yet they exhibit pseudo-random and ergodic behavior over time. These properties allow chaotic sequences to generate more irregular and better-distributed search trajectories than conventional uniform random perturbations. In the present study, the Chebyshev chaotic map is incorporated into GEO to construct the proposed CGEO. This section explains the role of chaos in metaheuristic optimization, introduces the Chebyshev map used in this study, and describes how the chaotic sequence is integrated into the GEO framework (Mohammadi-Balani *et al.*, 2021).

2.1 Role of chaos in metaheuristic optimization

In many metaheuristic algorithms, random numbers are used to control movement, perturbation, solution generation, or the balance between exploration and exploitation. In principle, these random values help diversify the population and prevent deterministic stagnation. However, when the search space is complex and contains many locally competitive solutions, conventional random sampling may not always provide sufficient search diversity. In such cases, agents can cluster too quickly around promising but suboptimal areas, especially when the exploitation component becomes dominant. Chaotic dynamics offers an alternative mechanism for diversification. Although chaotic sequences are generated by deterministic equations, they behave in an irregular and non-repetitive manner, making them suitable for replacing or modulating the random terms used in optimization algorithms (Eluri and Devarakonda, 2022; Yang *et al.*, 2023). The key advantage of chaos in this context is that it can improve search-space coverage while preserving a structured nonlinear search trajectory. Let a general chaotic map be represented as shown in Equation (5):

$$u_{t+1} = f(u_t) \quad (5)$$

where u_t is the chaotic state at iteration t , and $f(\cdot)$ is a nonlinear mapping function defined over a bounded interval. In most applications, the resulting sequence is then normalized into $[0, 1]$ so that it can directly replace the random factors used in the update equations of the optimizer.

In wrapper-based feature selection, the role of chaos is particularly important for three reasons. First, it can improve the initial diversity of candidate feature subsets. Second, it can diversify the movement of agents during iterative search, reducing repeated or synchronized trajectories. Third, it can help delay premature convergence by maintaining a broader balance between exploration and exploitation over time. These properties are consistent with the experimental findings presented in Overall Performance Comparison, Impact of Chaotic Chebyshev Map Integration, Statistical Robustness and Stability, and Feature-Selection Efficiency, where the proposed CGEO shows improved classification performance, stronger feature-reduction efficiency, and lower variability across repeated runs. Therefore, in this study, chaos is not introduced as a separate optimization principle, but as a dynamic search-enhancement mechanism embedded into the GEO framework.

2.2 Chebyshev map

Among the available chaotic maps, the Chebyshev map was selected as the chaotic generator for the proposed CGEO. The choice of this map is motivated by its strong ergodic behavior, low computational cost, deterministic irregularity, and suitability for repeated numerical optimization. These properties are desirable in wrapper-based feature selection because the optimizer must repeatedly explore a combinatorial subset space while maintaining enough diversity to avoid premature convergence. The Chebyshev map used in this study is defined as shown in Equation (6):

$$u_{t+1} = \cos(k \cos^{-1}(u_t)) \quad (6)$$

where $u_t \in [-1,1]$ is the chaotic state at iteration t , and k is the order of the Chebyshev polynomial. Because the output of the map lies in the interval $[-1, 1]$, it is transformed into the interval $[0, 1]$ before being used in the search process, shown in Equation (7):

$$z_t = \frac{u_t + 1}{2} \quad (7)$$

where $z_t \in [0,1]$ is the normalized chaotic coefficient. The normalized coefficient z_t can then be used in place of, or jointly with, the conventional random coefficients in the update rules of GEO. A generalized form of chaotic modulation may be expressed as shown in Equation (8):

$$r_t^{(c)} = z_t \text{ or } r_t^{(c)} = \lambda z_t + (1 - \lambda)r_t \quad (8)$$

where $r_t^{(c)}$ is the chaos-modulated coefficient, z_t is the normalized Chebyshev value, r_t is a conventional random number, and $\lambda \in [0,1]$ controls the degree of chaotic influence. The rationale for choosing the Chebyshev map instead of simpler maps lies in its ability to generate highly irregular trajectories with very low computational overhead. This is especially useful in the present study because:

- the population size is relatively large ($N = 100$),
- the optimization is repeated 30 independent times,
- and the algorithm must evaluate feature subsets under four different classifiers.

Accordingly, the Chebyshev map serves as the nonlinear driver of the proposed CGEO and provides the dynamic mechanism that later contributes to the observed improvements in stability, feature compactness, and predictive performance reported in Results and Discussion and summarized in Conclusion and Future Work.

2.3 Integration into GEO

The integration of the Chebyshev chaotic map into GEO is performed in a way that preserves the original hunting-inspired structure of the optimizer while enhancing its dynamic search behavior. In this study, chaotic modulation is introduced into the GEO framework through three main components: population diversity enhancement, movement control, and search-balance regulation.

1) Chaotic initialization: The first role of the chaotic map is to improve the diversity of the initial population. Rather than relying solely on conventional random initialization, the Chebyshev sequence can be used to distribute candidate solutions more irregularly across the feasible search domain. This helps

reduce the risk of highly similar starting solutions and improves the initial coverage of the feature subset space. In the context of wrapper-based feature selection, this step is important because the initial distribution of eagles affects how broadly the algorithm explores candidate subsets before exploitation intensifies. A more diverse initial population increases the chance of encountering informative feature combinations early in the search.

2) Chaotic control of exploration and exploitation: The second and more important role of the chaotic map is to modulate the movement dynamics of GEO during the cruise and attack phases. In the original GEO, these behaviors depend on stochastic coefficients. In the proposed CGEO, the normalized Chebyshev sequence is used to replace or modulate those coefficients so that the search trajectories become deterministic but non-repetitive (Prosper chuks, 2022). This idea is mathematically implemented later in Mathematical Representation of the CGEO, where the chaotic coefficient is incorporated into the movement equation of each eagle. By doing so, CGEO preserves the original GEO structure while introducing more diversified and nonlinear movement behavior. This mechanism is expected to:

- broaden search-space coverage in the early phase,
- reduce agent clustering around local optima,
- and maintain effective exploration–exploitation balance throughout the iterations.

This interpretation is consistent with the results reported in Impact of Chaotic Chebyshev Map Integration, where CGEO demonstrates more stable feature-selection behavior and improved optimization characteristics than the original GEO.

3) Chaotic parameter scheduling: The third integration mechanism concerns parameter control. In the standard GEO, the balance between cruising and attacking is commonly adjusted through coefficients that vary over time. In the proposed CGEO, chaotic values can be used to modulate this adjustment process so that the transition between exploration and exploitation does not become overly smooth or overly predictable. This helps avoid synchronized movement patterns and supports a more adaptive search behavior across iterations. In practical terms, this integration is aligned with the experimental configuration reported later in Table 2, where GEO and CGEO share the same basic settings, with the only additional component being the chaotic modulation introduced in CGEO. The resulting optimizer is therefore directly comparable with GEO under the same population size, iteration limit, classifier settings, and evaluation protocol.

Relation to the proposed framework: Through these three integration mechanisms, the chaotic map becomes an embedded component of the GEO framework rather than a separate preprocessing tool (Al-Gburi and Kurnaz, 2022). Its role is to strengthen the search stage of wrapper-based feature selection, which is later evaluated in terms of:

- classification accuracy and specificity, as discussed in Overall Performance Comparison;
- feature-selection stability, as examined in Impact of Chaotic Chebyshev Map Integration;
- statistical robustness, as presented in Statistical Robustness and Stability;
- dimensionality reduction, as evaluated in Feature-Selection Efficiency; and

- computational trade-offs, as discussed in Clinical and Computational Significance.

In summary, the integration of the Chebyshev map into GEO is intended to improve population diversity, delay premature convergence, and enhance the balance between exploration and exploitation without changing the fundamental structure of the optimizer. The actual implementation of this idea is described in the next section, where the proposed CGEO approach is formulated mathematically and operationally.

3. The proposed CGEO approach

The proposed CGEO is developed as an enhanced variant of the original GEO by incorporating Chebyshev chaotic dynamics into its search mechanism. The main objective of this modification is not merely to replace conventional random numbers with chaotic values, but to improve the search behavior of GEO in wrapper-based feature selection, particularly in problems where many candidate feature subsets exhibit similar local fitness values. Under such conditions, the standard GEO may experience premature convergence and reduced population diversity, which can limit its ability to identify compact and discriminative feature subsets.

In the present study, the Chebyshev chaotic map is integrated into the GEO framework to modulate the search dynamics of the optimizer. This modification is intended to improve search-space coverage, maintain population diversity, and strengthen the transition between exploration and exploitation. The practical effectiveness of this enhancement is evaluated in Experimental Setup and Parameter Configuration, Results and Discussion, and Conclusion and Future Work based on classification performance, feature-subset compactness, stability across repeated runs, and computational cost.

3.1 Framework overview

The proposed CGEO-based framework consists of three tightly connected stages: feature selection, classifier-guided evaluation, and comparative performance assessment.

1) Feature selection using CGEO: In the proposed framework, CGEO is employed as a wrapper-based optimization method to search for an informative and compact subset of features from the hypertension dataset. Each eagle represents a candidate solution encoded as a binary feature vector, shown in Equation (9):

$$X_i = [x_{i1}, x_{i2}, x_{i3}, \dots, x_{id}] \quad (9)$$

where $x_{ij} \in \{0,1\}$, d is the total number of available features, and $x_{ij} = 1$ indicates that the j -th feature is selected, whereas $x_{ij} = 0$ indicates that it is excluded. In this representation, each eagle corresponds to one possible feature subset. The goal of the optimizer is to identify a subset that yields strong predictive performance while avoiding unnecessary dimensionality.

2) Integration with machine learning classifiers: To evaluate the predictive quality of each candidate feature subset, the selected features are passed to one of four classifiers: SVM, KNN, DT, and NB.

This wrapper mechanism allows the optimizer to receive direct feedback from classification performance rather than relying solely on filter-based criteria such as correlation or information gain. As a

result, the feature-selection process becomes classifier-guided and better aligned with the predictive objective of the study.

3) Comparative evaluation under identical conditions: to ensure fairness, GEO and CGEO are evaluated under the same experimental conditions, including:

- identical dataset and preprocessing steps,
- the same population size and maximum number of iterations,
- the same classifier settings,
- the same train–test partitioning and validation strategy,
- and the same performance measures.

In addition to the main GEO–CGEO comparison, an additional SVM-based optimizer benchmark was conducted to evaluate CGEO against PSO, GWO, and GA under the same preprocessing, classifier configuration, fitness function, population size, iteration limit, independent-run setting, and evaluation metrics. Each optimizer–classifier combination is executed over 30 independent runs, and the results are summarized using mean and sample standard deviation. This design makes it possible to assess not only average classification performance but also stability, feature-selection efficiency, and runtime overhead, as reported throughout Results and Discussion. Overall, the role of CGEO in this framework is to strengthen the optimization stage of wrapper-based feature selection, while the classifiers serve as external evaluators of the quality of the selected feature subsets.

3.2 Mathematical representation of the CGEO

The proposed CGEO preserves the main movement structure of the original GEO but replaces the ambiguous stochastic modulation with a single reproducible chaotic update rule. Let (x_t) denote the Chebyshev chaotic state at iteration (t) . The chaotic sequence is generated as Equation (10):

$$u_{t+1} = \cos(k \cos^{-1}(u_t)) \quad (10)$$

where $u_t \in [-1,1]$ is the chaotic state at iteration t , and k is the order of the Chebyshev polynomial. Because the output of the Chebyshev map lies in the interval $[-1, 1]$, it is linearly transformed into the interval $[0, 1]$ using the normalization defined earlier in Equation (7). Let z_t denote the normalized chaotic coefficient obtained from that transformation. In the original GEO, the eagle movement is determined by cruise and attack tendencies. In CGEO, these components are modulated by the normalized chaotic sequence. Let C_i^t denote the cruise component and A_i^t denote the attack component of the i -th eagle at iteration t . The chaos-modulated velocity of the eagle is expressed as shown in Equation (11):

$$V_i^t = z_t \cdot C_i^t + (1 - z_t) \cdot A_i^t \quad (11)$$

The updated position of each eagle is then computed as shown in Equation (12):

$$X_i^{t+1} = X_i^t + fl \cdot V_i^t \quad (12)$$

where fl is the flight-length coefficient controlling the movement step size. The purpose of chaotic modulation is to provide a deterministic but non-repetitive search pattern. In the early phase of

optimization, this mechanism can improve the coverage of the search space and reduce the clustering of search agents around locally attractive regions. In later iterations, it can still maintain sufficient diversity to delay premature convergence while allowing local refinement around promising feature subsets. Therefore, the expected advantage of CGEO lies in the search dynamics it induces rather than in a complete reformulation of the GEO structure. This mathematical formulation is consistent with the later experimental observations in Results and Discussion, where CGEO yields better classification performance, smaller variability across repeated runs, and stronger feature reduction than the original GEO.

3.3 Fitness function design

In the proposed wrapper-based framework, each candidate feature subset is evaluated through a classifier-dependent fitness function that jointly considers classification accuracy and subset compactness. The purpose of this design is to prevent the optimizer from selecting excessively large subsets that provide only marginal predictive gains. The fitness function is defined as shown in Equation (13):

$$Fitness = \alpha(1 - Acc) + \beta \left(\frac{N_s}{N_t} \right) \quad (13)$$

where:

- **Acc** is the classification accuracy obtained from the selected subset,
- N_s is the number of selected features,
- N_t is the total number of available features,
- α and β are weighting coefficients used to balance predictive performance and dimensionality reduction. In this study, the weighting coefficients are specified in Equation (14):

$$\alpha = 0.9, \beta = 0.1 \quad (14)$$

so that greater importance is given to predictive performance while still penalizing unnecessarily large feature subsets.

The objective of the optimizer is to minimize the fitness value. Under this formulation:

- Higher classification accuracy reduces the objective value, and
- selecting more features increases the objective value.

This ensures that the optimizer searches for feature subsets that are both accurate and compact. Such a formulation is particularly suitable for medical classification problems, where reduced dimensionality may improve interpretability, computational efficiency, and practical deployment. The same fitness formulation is consistently applied to both GEO and CGEO in Experimental Setup and Parameter Configuration, ensuring a fair comparison between the two algorithms. In addition, the results presented in Feature-Selection Efficiency, where CGEO reduces the feature subset size by approximately 50% while maintaining superior classification performance, directly reflect the effectiveness of this fitness design.

3.4 Algorithm workflow

The workflow of the proposed CGEO follows the general structure of the standard GEO while introducing chaotic dynamics at key stages of the optimization process. The complete procedure can be summarized in six major steps, as shown in **Algorithm 2**.

1) Parameter initialization: the algorithm begins by defining the core parameters, including:

- population size N ,
- maximum number of iterations T ,
- attack propensity,
- cruise propensity,
- flight length fl ,
- chaotic perturbation amplitude δ ,
- and the initial state of the Chebyshev map.

These settings are aligned with the experimental configuration later reported in Table 2 under Parameter Settings for GEO and CGEO.

2) Population generation: A population of eagles is generated to represent candidate feature subsets in the search space. Each eagle is encoded as a binary vector. The initial population is distributed across the feasible search region, while the chaotic map provides nonlinear modulation that improves diversity and reduces the risk of highly similar initial trajectories.

3) Fitness evaluation: Each eagle is evaluated using the fitness function defined in Equation (13), with the weighting coefficients specified in Equation (14). In the wrapper setting, this means that the feature subset selected by the eagle is passed to the embedded classifier, and the resulting classification accuracy is combined with the feature reduction term to compute the objective value. The best-performing solution is stored as the current prey or elite solution (Yang *et al.*, 2023).

4) Cruise and attack movement updating: At each iteration, the optimizer computes the cruise and attack tendencies of each eagle following the original GEO logic (Magesh *et al.*, 2023). In CGEO, however, the movement coefficients are modulated by the normalized chaotic value z_t , producing more diversified search trajectories. This stage determines how the eagle explores or exploits the feature subset space (Abbasi Fashami *et al.*, 2024).

5) Chaotic control updating: The chaotic state is updated iteratively using the Chebyshev map, and the resulting sequence is used to regulate the movement dynamics throughout the optimization process. This step is intended to preserve diversity and introduce nonlinear perturbations into search behavior. The practical effect of this mechanism is later reflected in Impact of Chaotic Chebyshev Map Integration, where CGEO demonstrates more stable feature-selection behavior and improved convergence characteristics.

6) Selection and termination: After position updating, all candidate solutions are reevaluated. If a new position produces a lower fitness value, it replaces the previous memory of that eagle. The algorithm continues until the maximum number of iterations is reached or the convergence condition is satisfied. The final output of CGEO is the best feature subset found and its corresponding fitness value.

Algorithm 2. The CGEO algorithm.

- 1: Initialize the population size of eagles (N), maximum number of iterations (T), attack coefficient (α), cruise coefficient (β), and control parameter (λ)
- 2: Initialize the chaotic variable $c_0 \in (-1,1)$ and the Chebyshev order $k \geq 2$
- 3: Generate the initial population of eagles $\{X_i\}_{i=1}^N$ randomly within the search space
- 4: Evaluate the fitness value of all eagles and set the best one as the prey (elite solution)
- 5: **While** (termination criterion is not satisfied) **do**
- 6: Update the chaotic variable using the Chebyshev map

$$c_{t+1} = \cos(k \cdot \arccos(c_t)) \quad // \text{ use } c_{t+1} \text{ as a deterministic chaotic number}$$
- 7: **For** each eagle $i = 1 \rightarrow N$ **do**
- 8: Generate chaotic exploration factor

$$r_1 = \frac{c_{t+1}+1}{2} \quad // \text{ map from } [-1,1] \text{ to } [0,1]$$
- 9: Generate chaotic exploitation factor

$$r_2 = rand() \times r_1$$
- 10: Compute the cruise (exploration) velocity using

$$V_c = \beta \cdot r_1 \cdot (X_{\text{prey}} - X_i)$$
- 11: Compute the attack (exploitation) velocity using

$$V_a = \alpha \cdot r_2 \cdot (X_{\text{prey}} - X_i)$$
- 12: Combine both movements using the weighting parameter

$$V_i = \lambda \cdot V_c + (1 - \lambda) \cdot V_a$$
- 13: Update the position of eagle

$$X_i^{\text{new}} = X_i + V_i$$
- 14: Apply boundary control to keep X_i^{new} within the lower and upper limits
- 15: Evaluate the fitness value of the updated eagle
- 16: **End for**
- 17: Sort all eagles according to their fitness values
- 18: Update the prey (global best) if a better eagle is found
- 19: Update the control parameter λ (e.g. linearly decreasing) to gradually shift from exploration to exploitation
- 20: **End while**
- 21: Output the best fitness value and the corresponding prey position found by CGEO

In summary, CGEO extends the standard GEO by incorporating Chebyshev chaotic dynamics into the search process. This modification is intended to improve population diversity, reduce premature convergence, and strengthen the transition between exploration and exploitation in wrapper-based feature selection (Eluri and Devarakonda, 2022). The impact of this enhancement is evaluated in Experimental Setup and Parameter Configuration, Results and Discussion, and Conclusion and Future Work in terms of classification performance, feature-subset compactness, statistical robustness, and computational cost.

4. Experimental setup and parameter configuration

This section describes the experimental setup used to evaluate the proposed CGEO algorithm in comparison with the standard GEO. The configuration was designed to ensure fairness, reproducibility, and consistency across all experiments. In particular, the same dataset, preprocessing procedures, classifier settings, train–test partitioning strategy, cross-validation protocol, and evaluation metrics were applied to both GEO and CGEO. Therefore, any observed differences in classification performance, feature subset compactness, and runtime can be more reasonably attributed to the chaotic enhancement introduced in CGEO rather than to inconsistencies in the experimental design (Eluri and Devarakonda, 2022; Mohammadi-Balani *et al.*, 2021).

All experiments were conducted using MATLAB R2019a on a Windows 11 Pro operating system with an Intel® Core™ i9 processor, 64 GB RAM, and a 64-bit architecture. Each optimizer–classifier combination was executed for 30 independent runs. The repeated-run setting was used to reduce the influence of initialization randomness and support the stability analysis presented in Statistical Robustness and Stability. The present section is organized into five parts: dataset description, parameter settings, classifier configuration, evaluation metrics, and the overall experimental workflow.

4.1 Dataset description

The experimental evaluation was conducted using the hypertension_data.csv file extracted from the public dataset “Diabetes, Hypertension and Stroke Prediction” published by Prosper Chuks on Kaggle. In this study, only the hypertension-related subset was used for wrapper-based feature selection and classification. The dataset consists of anonymized clinical and demographic attributes represented in tabular form and was used to assess the effectiveness of GEO and CGEO under the same feature-selection framework.

Dataset source: Prosper Chuks, (2022). Diabetes, Hypertension and Stroke Prediction. Kaggle. Link: <https://www.kaggle.com/datasets/prosperchuks/health-dataset> The dataset contains 26,083 samples and 14 variables, including 13 predictor attributes and 1 target variable. The target variable was defined as 0 = non-hypertensive and 1 = hypertensive, following the coding used in the file analyzed in this study. The predictor variables include demographic information, blood pressure-related indicators, biochemical markers, and exercise- or ECG-related factors commonly used in cardiovascular risk analysis. Details are shown in Table 1.

Table 1 Description of the dataset and variables used in this study.

No.	Variable	Description	Type	Range / Coding
1	age	Age of the subject	Integer	11 - 98
2	sex	Gender	Binary	0 = Female, 1 = Male
3	cp	Chest pain type	Categorical	0 - 3
4	trestbps	Resting blood pressure (mmHg)	Numeric	94 - 200
5	chol	Serum cholesterol (mg/dL)	Numeric	126 - 564
6	fbs	Fasting blood sugar > 120 mg/dL	Binary	0 = No, 1 = Yes
7	restecg	Resting electrocardiographic result	Categorical	0 - 2
8	thalach	Maximum heart rate achieved	Numeric	71 - 202
9	exang	Exercise-induced angina	Binary	0 = No, 1 = Yes
10	oldpeak	ST depression induced by exercise	Float	0.0 - 6.2
11	slope	Slope of the peak exercise ST segment	Categorical	0 - 2
12	ca	Number of major vessels colored by fluoroscopy	Integer	0 - 4
13	thal	Thalassemia	Categorical	0 - 3
14	target	Diagnosis result	Binary	0 = non-hypertensive, 1 = hypertensive

The dataset was processed using a uniform and leakage-controlled preprocessing pipeline. Predictor variables already represented by numerical category codes were retained in their encoded form, whereas continuous variables were standardized using z-score normalization computed from the training data only. The same transformation parameters were then applied to the validation or test data. Missing values, if present, were handled within the training partition before model evaluation, using the median for continuous variables and the mode for categorical variables. Extreme values were inspected using distributional summaries and retained when they remained within plausible dataset-defined or clinically interpretable ranges, because such observations may represent high-risk cases rather than measurement errors.

For model evaluation, the dataset was divided into 70% training and 30% testing subsets using stratified sampling to preserve the original class distribution. Within the training portion, 10-fold stratified cross-validation was applied to evaluate candidate feature subsets generated by GEO and CGEO through the embedded classifiers. Each optimizer–classifier combination was executed for 30 independent runs using different optimizer seeds under the same data partitioning protocol. The final selected feature subset was then assessed on the unseen testing data. The same preprocessing steps, train–test partitioning, cross-validation setting, classifier configuration, and evaluation metrics were applied to both GEO and CGEO to reduce information leakage and ensure a fair comparison.

4.2 Parameter settings for GEO and CGEO

To ensure a fair comparison, the same optimization parameters were used for GEO and CGEO, except for the chaotic control component introduced in CGEO (Mohammadi-Balani *et al.*, 2021; Yang *et al.*, 2023). The settings were selected based on the recommendations of previous GEO-related studies and

were further adjusted through pilot experiments to provide a suitable balance between global exploration and local exploitation.

Table 2 Parameter settings of GEO and CGEO.

Parameter	Symbol	Value / Range	Purpose / Description	Applied in
Population size	N	100	Number of candidate solutions (eagles)	GEO, CGEO
Maximum iterations	T_{max}	100	Number of optimization cycles	GEO, CGEO
Attack propensity	α_t	$0.5 \rightarrow 2.0$	Controls exploitation intensity	GEO, CGEO
Cruise propensity	β_t	$1.0 \rightarrow 0.5$	Controls exploration intensity	GEO, CGEO
Flight length	fl	0.7	Step-size coefficient for position update	GEO, CGEO
Chaotic map type	–	Chebyshev map	Generates deterministic chaotic values	CGEO
Chaos perturbation amplitude	δ	0.05	Controls chaotic modulation intensity	CGEO
Fitness weights	α, β	0.9, 0.1	Accuracy–feature reduction trade-off	GEO, CGEO
Termination criterion	–	Max iteration / convergence	Stopping rule	GEO, CGEO

Both optimizers used the same population size, iteration limit, flight length, and weighting coefficients in the fitness function. The attack and cruise coefficients were dynamically adjusted during the search process, allowing greater exploration at the beginning of optimization and greater exploitation in later iterations. In CGEO, these movement-related components were additionally modulated by normalized chaotic values generated from the Chebyshev map, thereby introducing deterministic irregularity into the movement trajectories. As demonstrated in Impact of Chaotic Chebyshev Map Integration, this chaotic enhancement contributed to more stable feature-selection behavior and improved classification performance. Therefore, the parameter configuration in Table 2 is directly linked to the subsequent analyses of classification performance, stability, feature-reduction efficiency, optimizer benchmarking, and runtime overhead reported in Results and Discussion.

To support the controlled benchmark presented in Comparative Benchmark with Feature-Selection Optimizers, three additional feature-selection optimizers, namely PSO, GWO, and GA, were implemented under the same wrapper-based SVM setting. For fairness, these optimizers used the same dataset, preprocessing pipeline, 70:30 stratified train–test partition, 10-fold stratified cross-validation within the training set, fitness function, population size, maximum number of iterations, number of independent runs, and evaluation metrics as GEO and CGEO. The SVM configuration was also kept fixed across all optimizers. Thus, the benchmark was designed to compare the search behavior of the optimizers rather than differences caused by classifier tuning or data handling. For the additional benchmark optimizers, the algorithm-specific control settings were configured according to standard implementations commonly used in wrapper-based feature-selection studies. For PSO, the inertia weight was set to ($w = 0.7$), and the cognitive and social acceleration coefficients were set to ($c_1 = 1.5$) and ($c_2 = 1.5$), respectively. For GWO, the control parameter (a) was linearly decreased from 2 to 0 over the maximum number of iterations. For GA, a binary chromosome representation was used, with a crossover rate of 0.8 and a mutation rate of 0.01. The population size, maximum number of iterations, number of independent runs, fitness function, SVM

configuration, validation protocol, and evaluation metrics were kept identical across GEO, CGEO, PSO, GWO, and GA to ensure that the benchmark comparison reflected differences in optimizer search behavior rather than differences in experimental settings.

4.3 Classifier configuration

To evaluate the quality of the feature subsets selected by GEO and CGEO, four widely used supervised classifiers were embedded within the wrapper-based optimization framework: SVM, KNN, DT, and NB, as shown in Table 3. These classifiers were selected because they represent distinct learning paradigms and therefore provide a broader view of the generalization capability of the selected feature subsets. Each candidate solution generated by the optimizer was interpreted as a binary feature mask. The selected subset was passed to the classifier, and the resulting classification performance was returned to the optimizer in the form of a fitness value.

1) SVM (Cortes and Vapnik, 1995; Tanveer *et al.*, 2024) is a margin-based classifier designed for nonlinear and high-dimensional classification problems. In this study, the radial basis function (RBF) kernel was used because it can represent nonlinear relationships among clinical variables. The kernel is defined as shown in Equation (15):

$$K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2) \quad (15)$$

where γ controls the influence of each training sample in the transformed feature space. The SVM was configured as follows:

- Kernel: RBF
- Regularization parameter: $C = 10$
- Kernel coefficient: $\gamma = 0.01$
- Cross-validation folds: 10

SVM was the best performing classifier in the results reported in Overall Performance Comparison, particularly when combined with CGEO.

2) KNN (Halder *et al.*, 2024) is an instance-based classifier that predicts the class of a sample based on the majority vote among its nearest neighbors. Euclidean distance was used to measure similarity between observations, shown in Equation (16):

$$(x_i, x_j) = \sqrt{\sum_{k=1}^m (x_{ik} - x_{jk})^2} \quad (16)$$

where m is the number of selected features. The KNN configuration was:

- Number of neighbors: $k = 5$
- Distance metric: Euclidean
- Weighting scheme: Uniform
- Cross-validation folds: 10

Because KNN is sensitive to dimensionality, the quality of the selected feature subset directly influences its performance. This characteristic makes KNN suitable for evaluating the effectiveness of wrapper-based feature selection.

3) DT (Costa and Pedreira, 2023) is a rule-based classifier that recursively partitions the input space according to the attribute that maximizes node purity. The Gini impurity was used as the splitting criterion, shown in Equation (17):

$$Gini = 1 - \sum_{i=1}^c p_i^2 \quad (17)$$

where p_i is the proportion of observations in class i at a given node. The DT settings were:

- Splitting criterion: Gini impurity
- Maximum tree depth: 10
- Minimum samples per split: 2
- Pruning method: Cost-complexity pruning
- Cross-validation folds: 10

Although DT did not outperform SVM in terms of accuracy, it remained attractive because of its interpretability and strong feature reduction performance under CGEO, as demonstrated in Feature-Selection Efficiency.

4) NB (Albreiki *et al.*, 2024; Appasani *et al.*, 2024) is a probabilistic classifier based on Bayes' theorem under the conditional independence assumption. Since the preprocessed variables were treated numerically, the Gaussian version of NB was used, shown in Equation (18):

$$P(x | c) = \frac{1}{\sqrt{2\pi\sigma_c^2}} \exp \left(-\frac{(x-\mu_c)^2}{2\sigma_c^2} \right) \quad (18)$$

where μ_c and σ_c^2 are the class-specific mean and variance, respectively. The NB settings were:

- Distribution type: Gaussian
- Smoothing factor: 1
- Cross-validation folds: 10

Although NB is relatively simple, it serves as an important baseline model because its performance is highly dependent on the discriminative strength of the selected features.

5) Integration with GEO and CGEO: For both GEO and CGEO, the selected features were assessed using a classifier-dependent fitness function, shown in Equation (13). This formulation ensures that the optimizer seeks a compact subset with strong predictive performance. The use of the same fitness function and the same classifier settings for both GEO and CGEO guarantees a fair comparison.

Table 3 Classifier settings used in the wrapper-based framework.

Classifier	Learning paradigm	Core model setting	Key hyperparameters
SVM	Margin-based learning	RBF kernel	($C = 10$), ($\gamma = 0.01$)
KNN	Instance-based learning	Euclidean distance	($k = 5$), uniform weighting
DT	Rule-based learning	Gini impurity	Max depth = 10, min split = 2
NB	Probabilistic learning	Gaussian distribution	Smoothing factor = 1

The classifier hyperparameters in Table 3 were fixed a priori and applied identically to GEO and CGEO to isolate the effect of the feature-selection optimizer. The aim was not to obtain globally optimal classifier settings, but to determine whether the proposed chaotic enhancement improved the quality of the selected feature subsets under controlled classifier conditions. For SVM, the RBF kernel with ($C = 10$) and ($\gamma = 0.01$) was used to accommodate nonlinear patterns while controlling overfitting. For KNN, ($k = 5$) with Euclidean distance was selected to preserve local neighborhood information after standardization of continuous variables. For DT, the maximum depth was limited to 10 to reduce excessive tree growth while allowing nonlinear feature interactions. Gaussian NB was included as a simple probabilistic baseline whose performance depends strongly on the discriminative quality of the selected features.

Because the same hyperparameters were used for both GEO and CGEO, the observed performance differences can be attributed more directly to the feature-selection process rather than classifier tuning. Nevertheless, these settings should be interpreted as controlled experimental settings, not as globally optimal parameters. Further studies should examine CGEO under systematic hyperparameter tuning, such as grid search or nested cross-validation. This classifier configuration supports the comparative findings reported later in Overall Performance Comparison, where CGEO improves performance consistently across all four learning paradigms.

Because SVM achieved the strongest overall performance in the main GEO–CGEO comparison, it was selected as the fixed classifier for the additional optimizer benchmark reported in Comparative Benchmark with Feature-Selection Optimizers. This design allowed the benchmark to focus on the effect of the feature-selection optimizer while holding the classifier constant.

4.4 Evaluation metrics

To evaluate the performance of the selected feature subsets and the resulting classifiers, five standard performance measures were employed: accuracy, precision, recall, F1-score, and specificity. These metrics were derived from the confusion matrix and were chosen because they provide complementary views of classification quality in a medical context.

1) Confusion matrix (Markoulidakis and Markoulidakis, 2024; Singh *et al.*, 2021).

	Predicted Positive	Predicted Negative
Actual Positive	True Positive (TP)	False Negative (FN)
Actual Negative	False Positive (FP)	True Negative (TN)

where:

- TP: Number of correctly identified hypertensive patients.

- TN: Number of correctly identified normotensive individuals.
- FP: Non-hypertensive individuals incorrectly labeled as hypertensive.
- FN: Hypertensive patients incorrectly labeled as normotensive.

2) Accuracy measures the proportion of correctly classified cases among all observations (Al-Gburi and Kurnaz, 2022; Singh *et al.*, 2021), shown in Equation (19):

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (19)$$

3) Precision reflects the proportion of predicted positive cases that are truly positive (Singh *et al.*, 2021), shown in Equation (20):

$$Precision = \frac{TP}{TP + FP} \quad (20)$$

4) Recall (Sensitivity): Recall indicates the ability of the model to correctly identify positive cases (Singh *et al.*, 2021), shown in Equation (21):

$$Recall = \frac{TP}{TP + FN} \quad (21)$$

5) F1-score: the F1-score provides a harmonic balance between precision and recall (Singh *et al.*, 2021), shown in Equation (22):

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (22)$$

6) Specificity measures the proportion of correctly identified negative cases (Al-Gburi and Kurnaz, 2022; Singh *et al.*, 2021), shown in Equation (23):

$$Specificity = \frac{TN}{TN + FP} \quad (23)$$

7) Mean and sample standard deviation: Because each experiment was repeated 30 times, the central tendency and variation of each metric were summarized using the sample mean and sample standard deviation. In accordance with the reviewer's recommendation, the sample standard deviation is denoted by s rather than σ , shown in Equation (24):

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i; s = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2} \quad (24)$$

where $n = 30$ is the number of repeated runs.

8) Statistical significance testing: To determine whether the differences between GEO and CGEO were statistically meaningful, paired statistical testing was applied to the repeated-run results. This approach was appropriate because both algorithms were evaluated under the same experimental protocol across 30 independent runs. The resulting p -values were used to assess whether the observed differences in performance metrics were statistically significant at the 0.05 level. These statistical results support the interpretation in Statistical Robustness and Stability, where classification performance, stability, and feature-selection efficiency are jointly discussed.

4.5 Experimental design summary

The overall experimental design was constructed to evaluate CGEO and GEO in a controlled and reproducible manner. The design followed a consistent workflow from preprocessing to repeated evaluation.

1) Experimental workflow overview: The complete workflow consisted of the following stages:

- Dataset preparation: The hypertension dataset was preprocessed through normalization, encoding, data screening, and train-test partitioning.
- Feature selection stage: GEO and CGEO independently searched for informative feature subsets using the same population size, iteration number, and fitness function.
- Classifier integration: Each selected subset was evaluated using SVM, KNN, DT, and NB under fixed hyperparameter settings.
- Repeated experimentation: Each optimizer-classifier pair was executed over 30 independent runs.
- Performance aggregation: Mean and sample standard deviation were computed for all five-evaluation metrics.
- Statistical comparison: GEO and CGEO were compared using paired significance testing.

2) Simulation results and comparative analysis: The comparative performance results are summarized in Table 4 and visualized in Figure 1. These values represent the average outcomes over 30 independent runs.

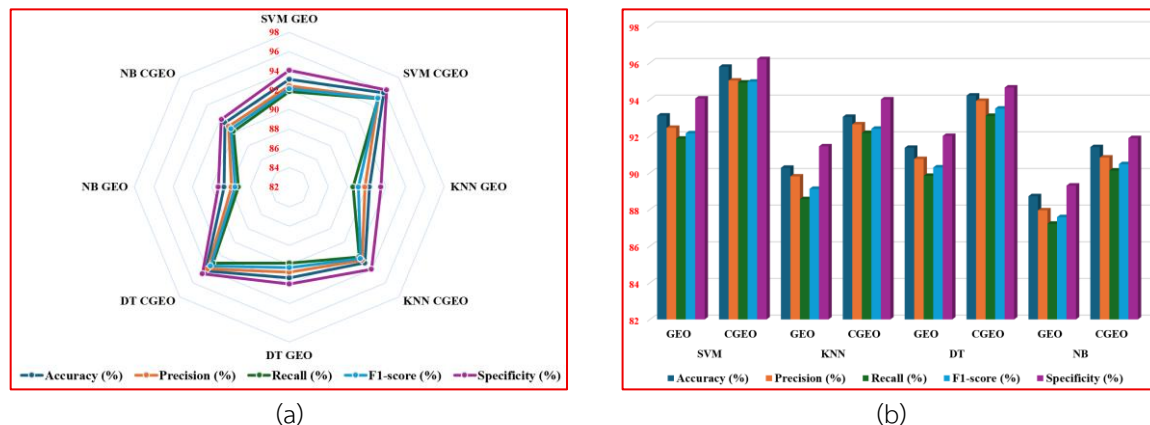


Figure 1 Comparative performance results of GEO and CGEO with SVM, KNN, DT, and NB across five evaluation metrics.

Table 4 Comparative performance and statistical significance of GEO and CGEO over 30 independent runs.

Classifier	Metric	GEO, mean $\pm$ SD (%)	CGEO, mean $\pm$ SD (%)	Test statistic	<i>p</i> -value
SVM	Accuracy	93.12 $\pm$ 0.84	95.78 $\pm$ 0.66	6.47	0.00001
	Precision	92.45 $\pm$ 0.97	95.03 $\pm$ 0.71	5.83	0.00002
	Recall	91.86 $\pm$ 1.10	94.92 $\pm$ 0.74	6.11	0.00001
	F1-score	92.15 $\pm$ 0.92	94.97 $\pm$ 0.69	6.24	0.00001
	Specificity	94.05 $\pm$ 0.77	96.20 $\pm$ 0.60	5.91	0.00002
KNN	Accuracy	90.27 $\pm$ 1.12	93.05 $\pm$ 0.95	4.86	0.00008
	Precision	89.80 $\pm$ 1.27	92.64 $\pm$ 0.89	4.65	0.00011
	Recall	88.56 $\pm$ 1.35	92.17 $\pm$ 1.02	5.03	0.00006
	F1-score	89.12 $\pm$ 1.19	92.40 $\pm$ 0.91	4.77	0.00009
	Specificity	91.44 $\pm$ 0.98	94.00 $\pm$ 0.84	4.52	0.00014
DT	Accuracy	91.36 $\pm$ 0.99	94.21 $\pm$ 0.73	5.22	0.00005
	Precision	90.75 $\pm$ 1.08	93.92 $\pm$ 0.79	5.04	0.00006
	Recall	89.83 $\pm$ 1.15	93.10 $\pm$ 0.84	4.97	0.00007
	F1-score	90.29 $\pm$ 1.04	93.50 $\pm$ 0.75	5.10	0.00006
	Specificity	92.01 $\pm$ 0.92	94.65 $\pm$ 0.68	4.81	0.00008
NB	Accuracy	88.72 $\pm$ 1.31	91.40 $\pm$ 1.02	4.54	0.00013
	Precision	87.95 $\pm$ 1.46	90.83 $\pm$ 1.08	4.36	0.00018
	Recall	87.22 $\pm$ 1.57	90.12 $\pm$ 1.20	4.21	0.00021
	F1-score	87.58 $\pm$ 1.41	90.47 $\pm$ 1.10	4.33	0.00019
	Specificity	89.30 $\pm$ 1.21	91.89 $\pm$ 0.97	4.16	0.00024

Note: Values are reported as mean $\pm$ sample standard deviation over 30 independent runs. The reported *p*-values were obtained from paired statistical comparisons between GEO and CGEO under identical experimental conditions.

As shown in Table 4 and Figure 1, CGEO consistently outperformed the original GEO across all classifiers and metrics. The lower variation observed in CGEO indicates improved stability, while the higher mean performance values suggest that the proposed optimizer was more effective in identifying discriminative feature subsets.

3) Summary of experimental implications: The experimental design directly supports the findings presented in Results and Discussion and summarized in Conclusion and Future Work. Specifically, the use of:

- repeated runs,
- identical classifier settings,
- common preprocessing,
- and the same fitness objective

ensured that the performance gains observed for CGEO can be interpreted as resulting from the chaotic enhancement rather than from uncontrolled differences in the experimental setup. Furthermore, the methodology presented in Experimental Setup and Parameter Configuration provides the basis for the conclusions drawn in Results and Discussion regarding:

- superior overall performance of CGEO,
- improved feature selection efficiency,
- stronger statistical robustness,
- and acceptable computational overhead.

RESULTS AND DISCUSSION

This section presents and discusses the experimental results obtained from the proposed CGEO in comparison with the original GEO. The discussion is organized into six parts: overall performance comparison, the impact of Chebyshev chaotic integration, statistical robustness and stability, feature selection efficiency, comparative interpretation with prior and broader benchmark methods, and the clinical and computational significance of the obtained results. All reported classification results were derived from 30 independent runs under identical experimental settings, and the values are presented as mean $\pm$ sample standard deviation.

1. Overall performance comparison

The overall comparative results of GEO and CGEO are summarized in Table 4 and visualized in Figure 1. Across all four classifiers, CGEO consistently achieved better classification performance than GEO in terms of accuracy, precision, recall, F1-score, and specificity. This pattern indicates that the chaotic enhancement improved the ability of GEO to identify more informative feature subsets and to produce more reliable classification outcomes. Among all evaluated combinations, CGEO-SVM yielded the best overall performance, achieving an accuracy of 95.78%, a precision of 95.03%, a recall of 94.92%, an F1-score of 94.97%, and a specificity of 96.20%. The CGEO-DT model also demonstrated strong performance, with an accuracy of 94.21%, suggesting that the proposed optimizer can improve both high-capacity classifiers and more interpretable tree-based classifiers. Likewise, CGEO improved the results of KNN and NB, although the magnitude of improvement was somewhat smaller than that observed for SVM and DT.

These results suggest that the proposed Chebyshev-chaos-enhanced search mechanism improved the wrapper-based feature selection process in a consistent manner across different learning paradigms. In addition to the higher mean values, the lower variation observed in CGEO indicates that the proposed optimizer is less sensitive to stochastic initialization and therefore offers improved experimental stability (Al-Gburi and Kurnaz, 2022).

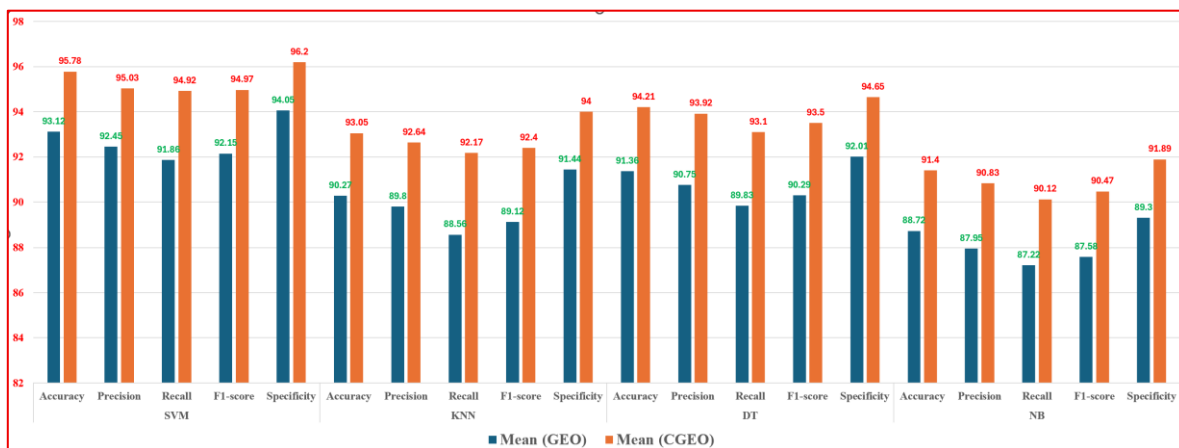


Figure 2 Comparative performance of GEO and CGEO across five evaluation metrics using SVM, KNN, DT, and NB.

Overall, the average performance gain of CGEO over GEO was approximately 2 - 3% depending on the classifier and metric considered. This magnitude of improvement, together with the reduction in standard deviation, indicates that the benefit of the proposed method is not limited to isolated runs, but reflects a more stable and effective optimization behavior (Al-Gburi and Kurnaz, 2022).

2. Impact of chaotic Chebyshev map integration

The key methodological distinction between GEO and CGEO is the incorporation of the Chebyshev chaotic map into the search dynamics of the optimizer. This modification was intended to improve search-space coverage, preserve population diversity, and reduce the risk of premature convergence. The empirical results indicate that this enhancement was beneficial in practice.

First, the performance gains observed in Table 4 suggest that CGEO was more effective than GEO in identifying discriminative feature subsets. Second, the lower sample standard deviations indicate that the proposed chaotic mechanism led to more stable solutions across repeated runs. These observations are consistent with the conceptual role of chaotic sequences in metaheuristic optimization, namely, to provide deterministic but irregular perturbations that diversify the movement of search agents. The impact of chaotic integration can also be interpreted through the consistency of selected features. To examine the stability of selected features, the selection frequency of each feature across the 30 independent runs was reported, as shown in Table 5. This analysis provides insight into the consistency of the feature-selection process and helps identify the core subset of variables repeatedly favored by CGEO (Boriratr et al., 2023).

Table 5 Frequency of feature selection by CGEO across 30 independent runs.

Feature	Selection frequency (%)	Interpretation
age	96.7	Frequently selected; reflects age-related cardiovascular risk
sex	71.3	Moderately important demographic factor
cp	93.3	Strongly informative clinical symptom indicator
trestbps	90.0	Frequently selected resting blood pressure variable
chol	86.7	Important biochemical risk factor
fbs	63.3	Selected in some runs; lower but still relevant
restecg	78.0	Moderately informative ECG-related factor
thalach	91.3	Strong indicator of cardiovascular response
exang	88.7	Important exercise-related symptom factor
oldpeak	95.3	One of the most consistently selected features
slope	84.7	Frequently selected ECG-derived factor
ca	92.0	Highly informative vascular feature
thal	89.3	Stable and important diagnostic feature

Table 5 shows that age, chest pain type (cp), resting blood pressure (trestbps), maximum heart rate achieved (thalach), exercise-induced ST depression (oldpeak), and number of major vessels (ca) were selected in the majority of runs, indicating that these variables formed a relatively stable core subset under the CGEO framework. In contrast, fasting blood sugar (fbs) and sex were selected less frequently, suggesting that their predictive contribution was comparatively weaker in this dataset. These findings indicate that CGEO did not improve predictive performance through arbitrary dimensionality reduction; rather, it tended to retain a consistent group of informative variables while excluding less stable or less discriminative predictors. This stable selection pattern supports the role of chaotic modulation in improving the search behavior of GEO and reducing sensitivity to initialization effects across repeated runs (Boriratrith *et al.*, 2023).

3. Statistical robustness and stability

To confirm that the observed improvements were not due to random variation, each optimizer-classifier combination was repeated 30 times, and the resulting performance metrics were subjected to statistical comparison. The repeated-run design allowed the central tendency and variability of each method to be summarized using the sample mean and sample standard deviation. In general, CGEO produced smaller standard deviations than GEO, indicating improved robustness to initialization effects.

The statistical comparison between GEO and CGEO is summarized in Table 4. Paired statistical testing was applied because both algorithms were evaluated under the same experimental protocol across repeated runs. The test was used to determine whether the performance differences between GEO and CGEO were statistically meaningful for each classifier and evaluation metric.

As shown in Table 4, all p -values were lower than 0.05, indicating that the observed differences between GEO and CGEO were statistically significant across all classifiers and metrics. The strongest

improvements were observed for SVM, which also yielded the highest overall classification performance. These results support the conclusion that Chebyshev chaotic enhancement improved the search effectiveness of GEO in a statistically reliable manner under the experimental conditions of this study.

4. Feature selection efficiency

An important objective of wrapper-based optimization is not only to improve classification performance but also to reduce the dimensionality of the selected feature subset. In this study, the proposed CGEO consistently selected fewer features than GEO while still achieving better predictive performance.

Table 6 Feature selection efficiency of GEO and CGEO.

Classifier	Algorithm	Average number of selected features	Reduction rate (%)
SVM	GEO	9.4	27.7
	CGEO	6.1	53.1
KNN	GEO	9.8	24.6
	CGEO	6.5	50.0
DT	GEO	9.1	30.0
	CGEO	5.9	54.6
NB	GEO	8.7	33.1
	CGEO	6.2	52.3

Table 6 shows that the average reduction rate of CGEO ranged from 50.0% to 54.6%, whereas GEO reduced the feature space by only 24.6% to 33.1%. The largest reduction was obtained with CGEO-DT, while CGEO-SVM provided the best trade-off between subset compactness and classification performance. These findings indicate that the proposed chaotic enhancement improved the optimizer's ability to eliminate redundant or weakly relevant features while preserving informative predictors. This outcome is particularly relevant to medical classification problems, where smaller feature subsets can improve model interpretability, reduce computational burden, and support more practical implementation.

Beyond dimensionality reduction, the selected feature subsets were also examined from a clinical interpretability perspective. The variables most frequently retained by CGEO, including age, chest pain type, resting blood pressure, maximum heart rate achieved, exercise-induced ST depression, number of major vessels, and thalassemia-related status, are consistent with cardiovascular risk assessment and hypertension-related classification. These variables reflect demographic risk, hemodynamic burden, exercise-related cardiac response, possible ischemic changes, vascular involvement, and cardiovascular comorbidity indicators. This suggests that CGEO did not reduce dimensionality arbitrarily; rather, it tended to preserve predictors with plausible medical relevance while removing less stable or redundant variables. Nevertheless, the selected features should be interpreted as predictive variables within a machine-learning framework, not as causal determinants of hypertension. Further validation using independent clinical datasets and expert review would be required before translating the selected feature subset into diagnostic

or screening practice. Given the improvement in both predictive performance and feature-subset compactness, an additional optimizer-level benchmark was conducted to further contextualize the competitiveness of CGEO relative to other commonly used feature-selection methods.

5. Comparative Benchmark with Feature-Selection Optimizers

The findings reported in the previous sections demonstrate that the proposed CGEO consistently improved the original GEO in terms of classification performance, feature-subset compactness, repeated-run stability, and computational trade-off. To provide an additional comparative context, this section presents a controlled SVM-based benchmark against three widely used feature-selection optimizers: PSO, GWO, and GA. These algorithms were selected because they represent common population-based and evolutionary search mechanisms frequently used in wrapper-based feature selection (Mohammadi-Balani *et al.*, 2021; Yang *et al.*, 2023).

To ensure a fair comparison, all optimizers in Table 7 were evaluated using the same hypertension dataset, preprocessing pipeline, 70:30 stratified train–test partition, 10-fold stratified cross-validation within the training set, SVM classifier configuration, fitness function, population size, maximum number of iterations, number of independent runs, and evaluation metrics. Therefore, the comparison focuses on algorithm-level differences in the feature-selection process rather than differences caused by classifier tuning, data handling, or validation protocol. The results are reported as mean $\pm$ standard deviation over 30 independent runs (Al-Gburi and Kurnaz, 2022; Mohammadi-Balani *et al.*, 2021).

Table 7 Controlled SVM-based benchmark comparison of feature-selection optimizers.

Optimizer	Classifier	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	Specificity (%)	Feature reduction (%)	Runtime (s)
GEO	SVM	93.12 $\pm$ 0.84	92.45 $\pm$ 0.97	91.86 $\pm$ 1.10	92.15 $\pm$ 0.92	94.05 $\pm$ 0.77	27.68 $\pm$ 3.39	31.53 $\pm$ 1.82
PSO	SVM	94.21 $\pm$ 0.76	94.08 $\pm$ 0.82	94.33 $\pm$ 0.74	94.19 $\pm$ 0.71	94.56 $\pm$ 0.69	39.22 $\pm$ 4.08	33.25 $\pm$ 2.04
GWO	SVM	94.67 $\pm$ 0.69	94.51 $\pm$ 0.75	94.82 $\pm$ 0.72	94.64 $\pm$ 0.68	95.03 $\pm$ 0.66	42.32 $\pm$ 3.79	34.14 $\pm$ 2.09
GA	SVM	93.86 $\pm$ 0.88	93.61 $\pm$ 0.93	94.02 $\pm$ 0.89	93.79 $\pm$ 0.86	94.18 $\pm$ 0.81	36.88 $\pm$ 4.47	35.69 $\pm$ 2.38
CGEO	SVM	95.78 $\pm$ 0.66	95.03 $\pm$ 0.71	94.92 $\pm$ 0.74	94.97 $\pm$ 0.69	96.20 $\pm$ 0.60	53.13 $\pm$ 2.87	34.83 $\pm$ 2.21

As shown in Table 7, CGEO achieved the highest overall classification performance among the optimizers compared under the SVM-based wrapper setting. Compared with the original GEO, CGEO improved accuracy from 93.12% to 95.78%, corresponding to an absolute gain of 2.66 percentage points. The proposed method also achieved the highest specificity, recall, and F1-score, suggesting a more balanced classification profile. In terms of feature selection, CGEO produced the most compact subset, with a feature-reduction rate of 53.13%, compared with 27.68% for GEO, 39.22% for PSO, 42.32% for GWO, and 36.88% for GA.

The runtime results indicate that CGEO required more computational time than GEO and PSO, but less time than GA. Although GWO showed a slightly lower runtime than CGEO, CGEO achieved higher predictive performance and stronger feature reduction. This suggests that the Chebyshev chaotic modulation introduced a moderate computational overhead but provided a more favorable balance between accuracy, specificity, feature-subset compactness, and runtime. Therefore, the results in Table 7

support the methodological premise that chaotic modulation can strengthen the search behavior of GEO and improve its competitiveness among population-based feature-selection optimizers.

Nevertheless, this benchmark should be interpreted within the scope of the present experimental setting. The comparison was conducted using a single hypertension-related dataset and a fixed SVM configuration. Therefore, the results should not be generalized to all datasets, classifiers, or parameter settings without further validation. Broader studies using additional clinical datasets, alternative classifiers, and systematic hyperparameter tuning are still required to confirm the robustness and generalizability of CGEO.

6. Clinical and computational significance

From a clinical perspective, the simultaneous improvement in recall and specificity is especially important. Higher recall indicates that the CGEO-based models were better able to correctly identify hypertensive individuals, thereby reducing false negatives. At the same time, higher specificity indicates that fewer non-hypertensive individuals were incorrectly classified as hypertensive, which is important for reducing unnecessary follow-up assessments and clinical burden. Thus, the proposed optimizer contributes not only to overall classification accuracy but also to a more balanced diagnostic profile. From a computational perspective, CGEO demonstrated two practical advantages. First, it improved predictive performance while reducing the average number of selected features by approximately half, which may support more compact and interpretable models. Second, although the chaotic enhancement introduced additional computational overhead, the runtime increase was relatively moderate in relation to the achieved gains in accuracy and feature reduction.

Table 8 Runtime comparison between GEO and CGEO.

Classifier	GEO runtime (s)	CGEO runtime (s)	Mean difference (s)	Interpretation
SVM	31.5 ± 1.8	34.8 ± 2.2	+3.3	Moderate overhead with best predictive performance
KNN	31.5 ± 1.6	34.8 ± 1.7	+3.3	Moderate increase with improved stability
DT	28.9 ± 1.4	32.1 ± 1.5	+3.2	Acceptable overhead for improved accuracy
NB	24.6 ± 1.2	27.3 ± 1.3	+2.7	Smallest runtime increase

Table 8 shows that CGEO required slightly more computation time than GEO for all classifiers, which is expected because the chaotic enhancement adds a dynamic modulation step to the optimization process. Nevertheless, the runtime increase remained relatively small, ranging from 2.7 to 3.3 seconds, whereas the performance and dimensionality-reduction gains were more substantial. Among the tested models, CGEO-SVM required the longest runtime but also achieved the best overall performance, while CGEO-NB showed the smallest runtime overhead.

Taken together, the results suggest that CGEO provides a practically meaningful improvement over GEO in both clinical and computational terms. The optimizer improved diagnostic discrimination, stabilized repeated-run behavior, and reduced feature dimensionality at an acceptable computational cost. These

characteristics make CGEO a promising candidate for feature-selection-driven medical classification systems, particularly where both predictive quality and model compactness are important.

CONCLUSIONS

This study proposed a CGEO algorithm for wrapper-based feature selection and hypertension classification. The proposed method extends the original GEO by incorporating a Chebyshev chaotic map into the search mechanism in order to improve population diversity, reduce premature convergence, and strengthen the transition between exploration and exploitation. The resulting optimizer was integrated with four classifiers, namely SVM, KNN, DT, and NB, and evaluated under the same experimental settings for a fair comparison with the conventional GEO. The experimental results demonstrated that CGEO consistently outperformed GEO across all evaluated classifiers and performance metrics. In particular, the CGEO-SVM model achieved the best overall performance, with the highest accuracy, precision, recall, F1-score, and specificity among all tested GEO–CGEO combinations. The additional controlled SVM-based benchmark further showed that CGEO achieved the most favorable overall trade-off among GEO, PSO, GWO, GA, and CGEO, particularly in terms of accuracy, specificity, F1-score, feature-reduction rate, and acceptable runtime.

In addition to the improvement in predictive performance, CGEO also reduced the dimensionality of the selected feature subsets more effectively than GEO, indicating that the proposed optimizer was able to identify more compact and informative feature subsets. The repeated-run analysis further showed that CGEO produced lower variability than GEO, suggesting greater robustness and stability in the optimization process. Overall, the findings indicate that the integration of Chebyshev chaotic dynamics into the GEO framework can improve both the search behavior of the optimizer and the classification quality of the selected feature subsets. Therefore, the proposed CGEO provides a promising enhancement of GEO for wrapper-based feature selection in medical data classification problems.

Theoretical and Practical Implications

From a theoretical perspective, this study contributes to the development of chaos-enhanced metaheuristic optimization by demonstrating how a Chebyshev chaotic map can be incorporated into the Golden Eagle Optimizer to influence search dynamics in a structured manner. The results suggest that the use of deterministic but non-repetitive chaotic modulation can improve the diversity of candidate solutions and stabilize the search process in wrapper-based feature selection. In this sense, the proposed CGEO extends the methodological scope of GEO and provides a useful reference for future studies that seek to combine chaotic dynamics with population-based optimization algorithms. The additional controlled benchmark also suggests that the proposed chaotic enhancement improves not only the baseline GEO but also the competitiveness of the optimizer relative to other population-based feature-selection methods under the SVM-based setting.

From a practical perspective, the proposed framework offers potential value for medical data analysis and decision-support applications. The improved classification performance suggests that CGEO can support more reliable identification of hypertensive cases, while the reduced feature subset size may

improve interpretability and computational efficiency. These characteristics are especially relevant in healthcare settings where compact predictive models, lower computational burden, and clinically meaningful feature subsets are desirable. In addition, the flexibility of the CGEO framework means that it may be adapted to other biomedical classification problems that require both dimensionality reduction and robust predictive performance.

Limitations and Future Work

Despite the promising results, several limitations should be acknowledged. First, although an additional controlled SVM-based benchmark was conducted against PSO, GWO, and GA, the comparison was still limited to a single hypertension-related dataset, a fixed SVM configuration, and the parameter settings adopted in this study. Therefore, the benchmark should be interpreted as evidence of CGEO's competitiveness under the present experimental setting rather than definitive proof of superiority across all feature-selection scenarios. Future studies should extend the comparison to additional datasets, alternative classifiers, different parameter settings, and real-world clinical environments. Second, the classifier hyperparameters were fixed to ensure a controlled comparison between GEO and CGEO. Although this design helped isolate the effect of the feature-selection optimizer, the selected hyperparameters should not be regarded as globally optimal for this dataset. Future studies should examine the robustness of CGEO under systematic hyperparameter tuning, such as grid search, randomized search, or nested cross-validation. Third, the present evaluation was based on a single public hypertension-related dataset. Although the dataset provides a useful testbed for classification, its source, sampling characteristics, and potential clinical bias may limit the generalizability of the findings. External validation using independent, multi-center, or real-world clinical datasets is therefore required before making stronger claims regarding practical deployment in clinical settings. Fourth, the current study focused on four conventional classifiers: SVM, KNN, DT, and NB. Future work should investigate whether the benefits of CGEO remain consistent across more advanced predictive models, including Random Forest, gradient boosting methods, and deep learning architectures. Such analyses would help determine the scalability of the proposed method across different learning paradigms. Fifth, although the selected features were clinically plausible, the present study remains predictive rather than causal. The selected variables should therefore be interpreted as machine-learning predictors, not as causal determinants of hypertension. Future studies should integrate explainable AI techniques, expert clinical review, and subgroup-specific analyses to strengthen the interpretability and clinical relevance of the selected feature subsets.

Finally, the current fitness function considered classification accuracy and feature-subset compactness. Future work may extend this formulation into a multi-objective optimization framework by incorporating additional criteria such as multicollinearity, computational cost, model interpretability, feature stability, and class-imbalance sensitivity. These extensions would help establish the robustness, scalability, and practical relevance of CGEO for broader medical data analysis tasks.

REFERENCES

- Abbasi Fashami, S., Zangian, M., Minuchehr, A. and Zolfaghari, A. (2024). Loading pattern optimization of VVER-1000 reactor core based on the discrete golden eagle optimization algorithm. *Nuclear Engineering and Technology* 56(8): 3425 - 3434. doi: 10.1016/j.net.2024.03.042.
- Albreiki, S., Simsekler, M.C.E., Qazi, A. and Bouabid, A. (2024). Assessment of the organizational factors in incident management practices in healthcare: A tree augmented Naive Bayes model. *PLoS ONE* 19(3): 1 - 20. doi: 10.1371/journal.pone.0299485.
- Al-Gburi, Z.D.S. and Kurnaz, S. (2022). Optical disk segmentation in human retina images with golden eagle optimizer. *Optik* 271: 170103. doi: 10.1016/j.ijleo.2022.170103.
- Appasani, D., Bokkisam, C.S. and Surendran, S. (2024). An Incremental Naive Bayes Learner for Real-time Health Prediction. *Procedia Computer Science* 235: 2942 - 2954. doi: 10.1016/j.procs.2024.04.278.
- Boriratr, S., Fuangfoo, P., Srithapon, C. and Chatthaworn, R. (2023). Adaptive meta-learning extreme learning machine with golden eagle optimization and logistic map for forecasting the incomplete data of solar irradiance. *Energy and AI* 13: 100243. doi: 10.1016/j.egyai.2023.100243.
- Cortes, C., Vapnik, V. and Saitta, L. (1995). Support-Vector Networks. *Machine Learning* 20: 273 - 297. doi: 10.1007/BF00994018.
- Costa, V.G. and Pedreira, C.E. (2023). Recent advances in decision trees: an updated survey. *Artificial Intelligence Review* 56(5): 4765 - 4800. doi: 10.1007/s10462-022-10275-5.
- Eluri, R.K. and Devarakonda, N. (2022). Binary Golden Eagle Optimizer with Time-Varying Flight Length for feature selection. *Knowledge-Based Systems* 247. doi: 10.1016/j.knosys.2022.108771.
- Halder, R.K., Uddin, M.N., Uddin, M.A., Aryal, S. and Khraisat, A. (2024). Enhancing K-nearest neighbor algorithm: a comprehensive review and performance analysis of modifications. *Journal of Big Data* 11(1): 1 - 55. doi: 10.1186/s40537-024-00973-y.
- Heidari, A.A., Mirjalili, S., Faris, H., Aljarah, I., Mafarja, M. and Chen, H. (2019). Harris hawks optimization: Algorithm and applications. *Future Generation Computer Systems* 97: 849 - 872. doi: 10.1016/j.future.2019.02.028.
- Jayalekshmi, S. and Velusamy, R.L. (2023). Energy aware Load Balanced Clustering using Golden Eagle Optimization in MS based LR-WPAN. *Procedia Computer Science* 220: 812 - 819. doi: 10.1016/j.procs.2023.03.108.
- Liu, H., Zhou, M. and Liu, Q. (2019). An embedded feature selection method for imbalanced data classification. *IEEE/CAA Journal of Automatica Sinica* 6(3): 703 - 715. doi: 10.1109/JAS.2019.1911447.
- Magesh, T., Devi, G. and Lakshmanan, T. (2023). Improving the performance of grid connected wind generator with a PI control scheme based on the metaheuristic golden eagle optimization algorithm. *Electric Power Systems Research* 214: 1 - 12. doi: 10.1016/j.epr.2022.108944.
- Markoulidakis, I. and Markoulidakis, G. (2024). Probabilistic Confusion Matrix: A Novel Method for Machine Learning Algorithm Generalized Performance Analysis. *Technologies* 12(7): 1 - 23. doi: 10.3390/technologies12070113.

- Mohammadi-Balani, A., Dehghan Nayeri, M., Azar, A. and Taghizadeh-Yazdi, M. (2021). Golden eagle optimizer: A nature-inspired metaheuristic algorithm. *Computers and Industrial Engineering* 152. doi: 10.1016/j.cie.2020.107050.
- Prosper chuks. (2022). Diabetes, Hypertension and Stroke Prediction [Data set]. Kaggle. https://www.kaggle.com/datasets/prosperchuks/health-dataset?select=hypertension_data.csv. Retrieved date 5 August 2025.
- Rakotonirainy, R.G. and van Vuuren, J.H. (2020). Improved metaheuristics for the two-dimensional strip packing problem. *Applied Soft Computing Journal* 92: 1 - 13. doi: 10.1016/j.asoc.2020.106268.
- Singh, P., Singh, N., Singh, K.K. and Singh, A. (2021). Chapter 5 - Diagnosing of disease using machine learning. In K.K. Singh, M. Elhoseny, A. Singh, & A.A. Elngar (Eds.), *Machine Learning and the Internet of Medical Things in Healthcare* (pp. 89 - 111). Academic Press. doi: 10.1016/B978-0-12-821229-5.00003-3.
- Tanveer, M., Rajani, T., Rastogi, R., Shao, Y.H. and Ganaie, M.A. (2024). Comprehensive review on twin support vector machines. *Annals of Operations Research* 339(3): 1223 - 1268. doi: 10.1007/s10479-022-04575-w.
- World Health Organization. (2024, May 7). WHO results report 2023 shows notable health achievements and calls for concerted drive toward Sustainable Development Goals. <https://www.who.int/news/item/07-05-2024-who-results-report-2023-shows-notable-health-achievements-and-calls-for-concerted-drive-toward-sustainable-development-goals>. Retrieved date 27 July 2025.
- Yang, X., Zhen, L. and Li, Z. (2023). Binary golden eagle optimizer combined with initialization of feature number subspace for feature selection. *Knowledge-Based Systems* 282: 1 - 18. doi: 10.1016/j.knsys.2023.111109.

